Master Erasmus Mundus in Photonics Engineering,
Nanophotonics and Biophotonics.

Europhotonics

MASTER THESIS WORK

Sensitivity Matrix Implementation in the Monte Carlo Package MBioICFO for the Simulation of Photon Migration in Tissues

Niklas Gerdes

Supervised by Prof. Turgut Durduran, (ICFO)
Co - supervised by Prof. Uli Lemmer, (KIT)

Presented on 23rd of October, 2015

Registered at

Escola Tècnica Superior d'Enginyeria de Telecomunicació de Barcelona

I hereby declare that the present thesis is original work written by me alone and that I have indicated completely and precisely all aids used as well as all citations, whether changed or unchanged, of other theses and publications.

Ich versichere hiermit, die vorliegende Arbeit selbstständig angefertigt, alle benutzten Hilfsmittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Abänderungen entnommen wurde.

Niklas Gerdes Barcelona, 15.10.2015

Abstract

An extension of the Monte Carlo method in diffuse optics was developed. Diffuse optical technology measures light absorption and scattering in human tissue. A sensitivity matrix has to be constructed to obtain structural information. It contains the sensitivities of the detected signal to absorption or scattering changes in different regions of the tissue. The existing MBioICFO simulation package was extended to allow the construction of the sensitivity matrix for perturbations in absorption. As opposed to prior implementations, the sensitivity matrix is determined in one simulation run and for arbitrary geometries. The new method was verified with analytical solutions for homogeneous media with infinite and semi-infinite boundary conditions. The method also enables determination of the sensitivity matrix for different detection times and in geometries obtained from MRI measurements. The program has shown appropriate computational efficiency with acceptable runtimes.

Acknowledgments

First and foremost I want to thank Prof. Turgut Durduran for giving me the opportunity to work on this project in his research group. I owe my progress to his patience with someone completely new to the field of diffuse optics.
Both academically and personally, I am grateful for the experience I have had in the medical optics group at ICFO. The research group provided me a supportive and motivational work environment. I would like to point out the supervision and guidance I received from Dr. Johannes Johansson throughout my project. Nicolas Mateos also helped me a lot during the initial setup of the program.
Besides, I want to thank the whole ICFO community for making my time here so enjoyable. The numerous extraordinary personalities that one constantly gets to meet at ICFO provide the perfect environment for academic and personal growth.
I thank Prof. Uli Lemmer at KIT for his willingness to co-supervise this work.
I also want to express my deep gratitude to everyone making the Erasmus Mundus Europhotonics program possible. In the two years I have been part of this program, I have studied and lived in three different countries, built relationships with people from all over the world and expanded my understanding of science. This amazing opportunity I have been given inspires me for all my future work. I hope I can give something back.
I could not have done any of this without the people closest to me. You know who you are. You are constantly with me in my thoughts.

Contents

Chapter 1

Introduction

There are various techniques to image living tissue and "see" the human body in a way invisible to our bare eyes. In diffuse optics, near-infrared (NIR) light is used to obtain physiological information about biological tissues [8].
Photons in this spectral range ($\sim$650-900 nm) experience low absorption in water and hemoglobin and can therefore travel deep into tissue [27]. Starting from a light source on the surface, photons propagate through the tissue, are scattered multiple times and are ultimately either absorbed, leave the tissue, or are detected by a detector placed at short distance ($\sim$mm-cm) from the source. On their way from source to detector, these photons have experienced absorption, scattering and phase shifts from moving scatterers, all of which influence the detected signal. Hence, analysis of the signal provides information about tissue absorption, scattering and flow of scatterers.
Typical absorption lengths after which a photon is absorbed are several centimeters while scattering occurs at distances less than a millimeter. Main absorbers in biological tissue are water, melanin and hemoglobin. Scattering predominantly happens at cell nuclei and mitochondria, since the refractive index difference to the surrounding water or lipid is large. Diffuse optical technology can measure changes in absorption and scattering and therefore changes in the concentrations of absorbers and scatterers. This can be relevant, for example, to measure blood oxygenation [2]. Since oxy- and deoxyhemoglobin have different absorption spectra, measuring absorption at several wavelengths allows to determine their con-

centrations. This technique is called near-infrared spectroscopy (NIRS) and has become a valuable method to monitor the state of anesthesia during surgery [13]. Besides, measuring the flow of scatterers allows determination of relative blood flow, since red blood cells are major scatterers. Diffuse correlation spectroscopy (DCS) monitors relative blood flow which is physiologically relevant since it is a measure of cancer growth or brain activity [4]. Combining blood oxygenation and blood flow measurements, the important physiological parameter of metabolic rate of oxygen extraction can be recorded [6].

For tomographical purposes, multiple sources and detectors are used. To actually reconstruct a structural image, however, mathematical models, and especially information on where in the tissue the detected photons have traveled, are required. It is important to know how an inhomogeneity in absorption or scattering affects the detected signal depending on where in the material it is located [8].

This work is about how to obtain this information using numerical simulations, so-called Monte Carlo simulations. In the Monte Carlo method, the propagation of photons through biological tissue is simulated which is used to estimate a range of physical quantities such as diffuse reflectance [26]. The goal of this work is to modify the existing Monte Carlo simulation package MBioICFO such that it can compute the sensitivity of the detected signal to changes of optical properties in different volume elements (voxels) of the material. Since the Monte Carlo method is accurate but computationally inefficient, this computation should be done in a single execution of the program, as opposed to prior solutions that required two executions and hence twice as much time. The motivation is to enable the determination of sensitivities of any asymmetric tissue geometry previously obtained from magnetic resonance imaging (MRI) measurements.

The challenge is twofold. On the one hand, the implementation has to be computationally efficient, feasible even. On the other hand, the computational results have to be physically correct. Verification of the first aspect is rather trivial. Acceptable runtimes and memory usage have to be achieved for an appropriate number of simulated photons. To ensure moderate memory usage, I implemented a class to create sparse matrices that store information of a photon's path. This information is stored in a different matrix if the photon is detected, or deleted if the photon does not reach the detector. The memory taken up by the sparse matrix is then

freed and used for the next photon.

To verify physical correctness, the simulation results can be compared to an analytical solution. An analytical solution, however, can only be found for a homogeneous medium and is only accurate far away from source and detector [20], which also illustrates the importance of the numerical Monte Carlo solutions. The solution I suggest considers information about location of scattering events and absorption of each detected photon to derive the sensitivity of the signal. I lay out details of the solution in the following chapters of this thesis. The approach is to first fully understand the physics involved and subsequently develop an implementation which then has to be verified and optimized.

In the following thesis, I first present the theoretical background to understand the physics involved in the problem. A general introduction to diffuse optics is given first, along with a mathematical description of photon transport in biological tissues. Then I provide an introduction to image reconstruction and an explanation of the relevance of the sensitivity that we are trying to compute.

In the subsequent section, I present a thorough introduction to the Monte Carlo method. I portray its importance and explain what problems it solves. I base most of my explanations on the MBioICFO simulation package but also provide an overview over different Monte Carlo approaches and algorithms. This lays the foundation to develop a theoretical derivation of my approach to compute the sensitivity.

I present the results of that implementation in the subsequent section, along with a comparison to the analytical solution. The code is analyzed and verified in the infinite as well as the semi-infinite medium. The code's ability to determine the sensitivity at different detection times is shown. Its usefulness for heterogeneous media is demonstrated with an MRI of a human head.

Finally, I discuss the results, draw a conclusion and give an outlook to the future.

Chapter 2

Theoretical Background

2.1 Diffuse Optics

The field of diffuse optics exploits the physical reality that water and hemoglobin show very small absorption of light in the spectral range from 650 to 900 nm. In this so-called "physiological window" (see figure 2.1), light can penetrate several centimeters into biological tissue before being fully absorbed [8]. Light propagation is then characterized by high scattering and low absorption, meaning a photon experiences multiple scattering events before being absorbed. To acquire meaningful information from detected signals, a physical model was derived to describe photon transport in diffusive, so-called turbid media. Analytically, photon transport in biological tissue is modeled by the radiation transport equation (RTE) [28]. The quantity of interest is the light radiance, $L(\vec{r}, \hat{\Omega}, t)(W/cm^{-2}sr^{-1})$, defined as the energy flow per unit time per unit solid angle per unit normal area. It can readily be seen that it corresponds to the intensity traveling in the $\hat{\Omega}$ direction.

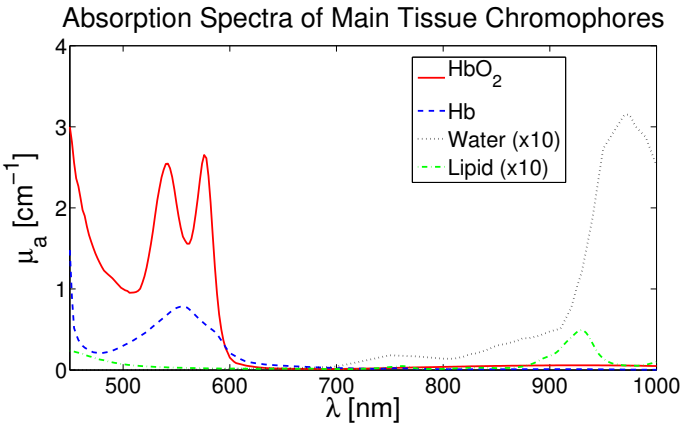

Figure 2.1: Illustration of "physiological window", the spectral range in which the main tissue chromophores show relatively low absorption [11, 17, 23]. The absorption coefficient μ_a is plotted vs. wavelength λ.

The radiation transport equation is derived from a consideration of conservation of energy. The left-hand side of the RTE as shown in equation 2.1 represents the change in energy per unit time. The right-hand side of the equation contains the four contributions to the energy change. The first term represents the energy loss due to divergence of the photon beam, the second one the extinction due to scattering and absorption, the third one the energy increase due to scattered photons and finally the fourth one describes the contribution of sources.

$$\begin{aligned}\frac{1}{v}\frac{\partial L(\vec{r},\hat{\Omega},t)}{\partial t} &= -\hat{\Omega}\cdot\nabla L(\vec{r},\hat{\Omega},t) - \mu_t L(\vec{r},\hat{\Omega},t) \\ &+\mu_s \int_{4\pi} L(\vec{r},\hat{\Omega}',t)P(\hat{\Omega}\cdot\hat{\Omega}')d\hat{\Omega}' + S(\vec{r},\hat{\Omega},t)\end{aligned} \tag{2.1}$$

v is the speed of light in the respective medium. The radiance $L(\vec{r},\hat{\Omega},t)$ depends on position $\vec{r}$, direction $\hat{\Omega}$ and time t. Absorption and scattering are formally characterized by the coefficients $\mu_a(cm^{-1})$ and $\mu_s(cm^{-1})$, respectively. The attenuation due to absorption and scattering is described by the total attenuation coefficient $\mu_t = \mu_a + \mu_s$. The product $P(\hat{\Omega}\cdot\hat{\Omega}')d\hat{\Omega}'$ gives the probability that a photon is scattered at a scattering angle whose cosine is $\hat{\Omega}\cdot\hat{\Omega}'$. The assumption is that the initial and final direction of the photon do not matter, only the angle

between them. At last, $S(\vec{r}, \hat{\Omega}, t)$, the source term, has the units $W/(cm^3 sr)$. It gives the power per volume element emitted by sources at position $\vec{r}$ at time t in $\hat{\Omega}$ direction.

With six independent variables, the RTE is very complex and difficult to solve. To reduce complexity, one key assumption is made, namely the diffusion approximation. The diffusion approximation assumes a much higher scattering than absorption. An important quantity here is the reduced scattering coefficient μ_s'. Unlike the scattering coefficient μ_s, it is a measure of the distance photons travel before their path is randomized. In general, scattering can be high, but if only forward scattering in the same direction occurs, the randomization of the photon path is minimal. The link between the two coefficients is the scattering anisotropy g. It is defined as the ensemble average of the cosine of the scattering angle θ, $g = \langle \cos\theta \rangle$. The reduced scattering coefficient is then defined as $\mu_s' \equiv \mu_s(1-g)$. The diffusion approximation is that $\mu_s' \gg \mu_a$. That is exactly the case for biological tissue in the "physiological window". It means that the distance a photon travels before its path is randomized, the transport mean-free path $l_{tr} = 1/\mu_s'$, is much smaller than the absorption length, $1/\mu_a$, the typical distance after which a photon is absorbed. This approximation allows to consider $L(\vec{r}, \hat{\Omega}, t)$ as nearly isotropic, at least far away from light sources and boundaries.

To simplify the RTE, the radiance is written as a series expansion of spherical harmonics. Within the diffusion approximation, however, only the first-order spherical harmonics are included in the expansion. In this way, the diffusion equation is derived (for a thorough derivation, see Wang et al. [28]):

$$\frac{1}{v}\frac{\partial \Phi(\vec{r}, t)}{\partial t} + \mu_a \Phi(\vec{r}, t) - \nabla \cdot [D \nabla \Phi(\vec{r}, t)] = S(\vec{r}, t) \tag{2.2}$$

The diffusion coefficient D is defined as

$$D \equiv \frac{1}{3(\mu_s' + \mu_a)} \tag{2.3}$$

Besides, the new quantity of interest in the diffusion equation is the fluence rate $\Phi(\vec{r}, t)(W/m^2)$, defined as the energy flow per unit area per unit time. As opposed to the radiance, it does not depend on flow direction. From the radiance, the

fluence rate is obtained in the following way:

$$\Phi(\vec{r},t) = \int_{4\pi} L(\vec{r},\hat{\Omega},t)d\Omega \tag{2.4}$$

To summarize, the approximations made to derive the diffusion equation are the following: a much bigger reduced scattering coefficient than absorption coefficient (a factor of at least 10), an isotropic source, slow temporal variations, photon propagation distances larger than the transport mean-free path, scattering angles independent of initial photon direction and observation points far from sources and boundaries.

The usual way to solve the diffusion equation is to employ the Green's function. This is the function that gives the fluence rate for an infinitely short-pulsed point source $S(\vec{r},t) = \delta(\vec{r},t)$. The form of the Green's function depends on the imposed boundary conditions. For an infinite, homogeneous medium, the Green's function in the time-domain is:

$$G(\vec{r},t) = \frac{v}{(4\pi Dvt)^{3/2}} \exp\left(-\frac{r^2}{4Dvt} - \mu_a vt\right) \tag{2.5}$$

This function describes the broadening due to scattering, except for the $\exp(-\mu_a vt)$ term which represents absorption. To find solutions for any arbitrary source $S(\vec{r}',t')$, we take the convolution of the Green's function and the respective source term:

$$\Phi(\vec{r},t) = \int_0^t \int_0^\infty G(\vec{r},t,\vec{r}',t')S(\vec{r}',t')d\vec{r}'dt' \tag{2.6}$$

The Green's function in the continuous wave regime is found by eliminating the time-derivative term on the left-hand side of the diffusion equation 2.2. The Green's function then reads:

$$G(\vec{r},\vec{r}') = \frac{1}{4\pi|\vec{r}-\vec{r}'|} \exp(-k|\vec{r}-\vec{r}'|) \tag{2.7}$$

In this equation, k is defined as $k \equiv \sqrt{\mu_a/D}$.

Obviously, one is interested in geometries that model biological tissues. These include boundaries which means that appropriate boundary conditions have to be found. One already very useful geometry is the semi-infinite medium, assuming

one boundary plane between turbid medium and for example air. To find boundary conditions, one considers that photons leaving the tissue will never re-enter it again. Hence, all incoming radiance is due to Fresnel reflections at the interface. From this consideration, one obtains the partial-flux boundary condition which is exact but cumbersome to handle. A more practical approximation of it is called the extrapolated-zero boundary condition:

$$\Phi(z = -z_b) = 0 \tag{2.8}$$

It is derived from a Taylor expansion of the fluence rate in the partial-flux boundary condition. It states that the fluence becomes zero at a plane parallel to the interface outside the turbid medium at $z = -z_b$, where the z-direction is perpendicular to the boundary plane with positive z-values inside the turbid medium. The distance $z_b = 2l_{tr}(1 + R_{eff})/3(1 - R_{eff})$ depends on l_{tr} and R_{eff}. Collimated beam sources at the surface are approximated as isotropic sources at a depth $z = l_{tr}$, when the photon paths are completely randomized. The coefficient R_{eff} depends on the refractive indices of turbid and adjacent medium. If there is no index mismatch, $R_{eff} = 0$. For a refractive-index-mismatched boundary with $n = n_{in}/n_{out}$, the effective reflection coefficient can be approximated as $R_{eff} \approx -1.440n^{-2} + 0.710n^{-1} + 0.668 + 0.00636n$.
From electrostatics, we know the method of image charges to fulfill different boundary conditions. In analogy to that, we can use image sources to fulfill the extrapolated-zero boundary condition. The idea is to introduce a second, negative source outside the turbid medium. Placing a negative point source at position $z_s = -(2z_b + l_{tr})$, the superposition of the two point sources makes the fluence vanish in the desired plane. The two point sources are themselves infinite medium solutions, but their superposition is the semi-infinite medium solution. Therefore, the Green's function in the continuous wave regime for the semi-infinite medium is:

$$G(\vec{r}, r_1, r_b) = \frac{1}{4\pi}\left(\frac{\exp(-kr_1)}{r_1} - \frac{\exp(-kr_b)}{r_b}\right) \tag{2.9}$$

This is obviously merely the sum of two point sources as depicted in equation 2.7. We assume that both sources lie in the origin of the xy-plane. Accordingly, polar

coordinates are a good choice of coordinate system with $\rho^2 = x^2 + y^2$ and the source position variables r_1 and r_b are:

$$r_1 = \sqrt{(z - l_{tr})^2 + \rho^2} \tag{2.10}$$

$$r_b = \sqrt{(z + 2z_b + l_{tr})^2 + \rho^2} \tag{2.11}$$

Measurements can be carried out with source-detector pairs either in transmission or reflection geometry. In reflection geometry, light is detected on the same surface where it was injected by the source, some distance ρ away. In transmission geometry, on the other hand, the detector is placed on a surface parallel to the one where the light is injected. For the diffusion equation to be valid, source-detector separation should correspond to at least three times the transport mean-free path l_{tr}. Tissue measurements are done with three different light sources: continuous wave (CW), time pulsed for time-resolved spectroscopy (TRS) and intensity modulated to measure in the frequency-domain (FD). CW sources provide a constant intensity and information is obtained from measuring the drop in intensity some distance ρ away from the source. They are simple and easy to handle, but μ_a and μ_s' cannot be determined simultaneously. Time pulsed sources emit a short light pulse of the order of less than 100 ps. Propagating through the medium, the pulse broadens and detected photons provide information about different locations in the medium depending on when they are detected. In TRS, both μ_a and μ_s' can be determined simultaneously. Intensity modulated sources, finally, contain the same information content as time pulsed sources. Thus, μ_a and μ_s' can also be found in a single measurement. The light intensity is modulated sinusoidally at angular frequencies of between 100 MHz to 1GHz. One determines the optical properties by recording the amplitude and phase changes. Time pulsed light and intensity modulated light are simply related via a Fourier transform.

2.2 Image Reconstruction and the Jacobian

The general purpose in tomography is to reconstruct a three-dimensional image of the object under investigation. In diffuse optics, there is a mathematical formalism

to reconstruct an image of the optical properties from fluence rate measurements of various source-detector pairs [2].
When traveling through tissue, photons move on a random walk with mean step size l_{tr}. The step size as well as the scattering angles obey probability distribution functions. So their actual values at a specific scattering event cannot be predicted exactly. A visualization of how a photon moves through a turbid medium can be seen in figure 2.2.

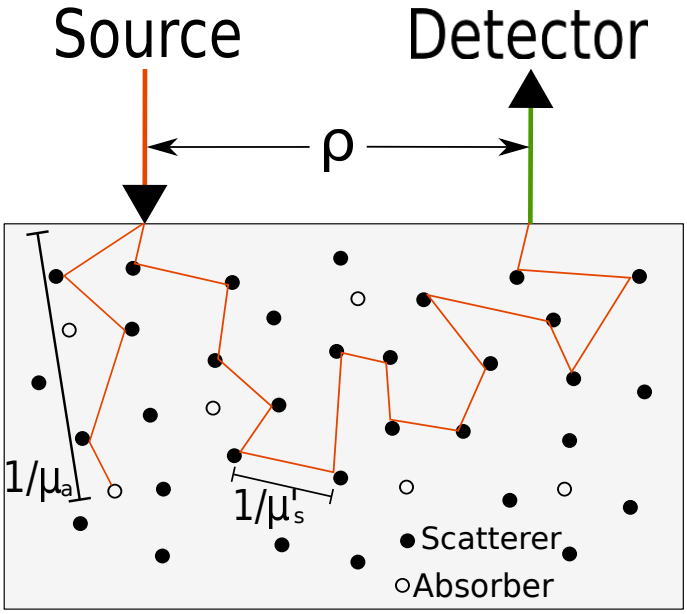

Figure 2.2: Illustration of photon random walk for a detected and an absorbed photon. Note that typically $1/\mu_a \gg 1/\mu_s'$ for turbid media in diffuse optics.

In figure 2.3, we can see the regions in the turbid medium that most detected photons have passed on their way from source to detector. For a source-detector pair in reflection geometry and with a certain distance ρ, this region has a banana-like shape. For zero source-detector separation, it resembles a drop.
Naturally, the optical properties of the tissue in these regions will affect the detected signal more than those in parts of the tissue that few or no detected photons have visited. To obtain structural information about the tissue from the measured signal, one needs to know how perturbations in the optical properties at different locations influence the fluence rate at the detector. In the following, I will present the theoretical framework of frequency-domain image reconstruction in an infinite

medium.

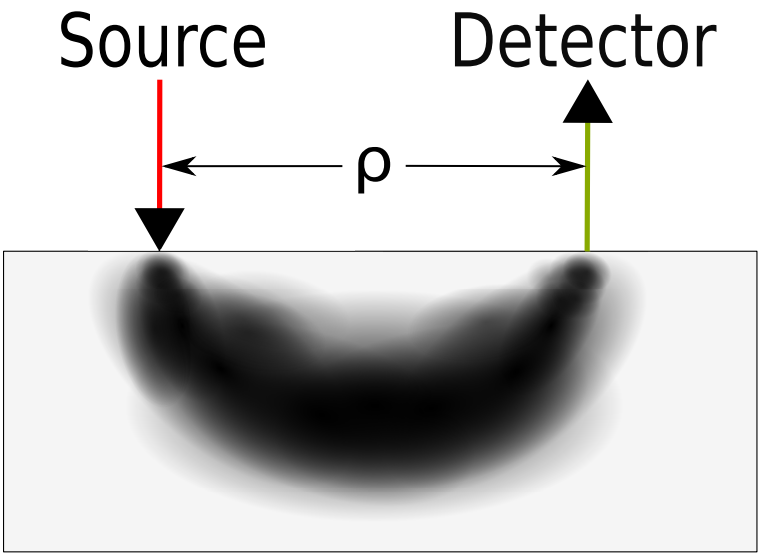

Figure 2.3: Illustration of path distributions of detected photons

We assume that the tissue we want to image has perturbations of the absorption coefficient μ_a only (equation 2.12). For contrast in the reduced scattering coefficient μ_s', the derivation is done in a similar fashion.

$$\mu_a(\vec{r}) = \mu_{a0} + \delta\mu_a(\vec{r}) \tag{2.12}$$

This means that there is a spatially dependent perturbation $\delta\mu_a(\vec{r})$ additional to the constant background absorption coefficient μ_{a0}. We assume the perturbation to be small compared to the background, $\delta\mu_a(\vec{r}) \ll \mu_{a0}$. The measured fluence rate $U(\vec{r})$ contains a part $U_0(\vec{r})$ caused by the background and a part $U_{sc}(\vec{r})$ caused by the perturbation. The Born approach formulates the fluence rate as $U(\vec{r}) = U_0(\vec{r}) + U_{sc}(\vec{r})$, whereas the Rytov approach states it as $U(\vec{r}) = U_0(\vec{r})\exp(U_{sc}(\vec{r}))$ [12]. We will focus on the Born approach. In the forward problem, the fluence rate change due to the perturbation in absorption is calculated. For image reconstruction from fluence rate measurements, however, the inverse problem has to be solved. That means that from the change of the fluence rate, the perturbation $\delta\mu_a(\vec{r})$ is determined. We approximate the fluence rate as a

first-order Taylor expansion:

$$U(\vec{r}) = U_0(\vec{r}) + \frac{\partial U_0(\vec{r})}{\partial \mu_a}\delta\mu_a \tag{2.13}$$

For practical reasons and since spatial resolution is low, the object to be imaged is divided into volume elements, so-called voxels. The fluence rate at the detector has a different sensitivity $\partial U_0/\partial \mu_a$ for each voxel. The matrix [W] containing the sensitivities of all voxels is called sensitivity matrix or Jacobian. It is the link between the perturbations in absorption and the fluence rate at the detector:

$$[U_{sc}(\vec{r})] = [W][\delta\mu_a(\vec{r})] \tag{2.14}$$

Equation 2.14 written out explicitly results in:

$$U_{sc}(\vec{r_{s_i}}, \vec{r_{d_i}}) = \sum_{j}^{N_V} W_{ij}\delta\mu_a(\vec{r}_j) \tag{2.15}$$

The index i refers to the source-detector pair while the index j refers to the respective voxel inside the medium. The position vectors $\vec{r}_s$ and $\vec{r}_d$ denote the location of source and detector, respectively, N_V is the number of voxels.
Determining the Jacobian numerically for any kind of asymmetric geometry using Monte Carlo simulations is the goal of this work. To verify the results though, we can set up an analytical expression of the Jacobian for a homogeneous medium. The approach is to plug the expressions for the fluence rate according to Born $U(\vec{r}) = U_0(\vec{r}) + U_{sc}(\vec{r})$ and expression 2.12 into the diffusion equation 2.2. This leads to:

$$(\nabla^2 - k^2)U_{sc}(\vec{r}) = \frac{\delta\mu_a(\vec{r})}{D}U(\vec{r}) \tag{2.16}$$

Following the usual Green's function approach, we can solve equation 2.16 by taking the convolution of the right-hand side and the corresponding Green's function:

$$U_{sc}(\vec{r}_s, \vec{r}_d) = \int \frac{-\delta\mu_a(\vec{r})}{D} G(\vec{r}_d, \vec{r})U(\vec{r}, \vec{r}_s)d^3r \tag{2.17}$$

Once again discretizing the object into voxels, we can write the integral as a sum

just as shown in equation 2.15. The sensitivities W_{ij} are then:

$$W_{ij} = \left[\frac{\partial U_0}{\partial \mu_a}\right]_{ij} = \frac{-\Delta V}{D} G(\vec{r}_{d_i}, \vec{r}_j) U_0(\vec{r}_j, \vec{r}_{s_i}) \tag{2.18}$$

The voxel size is denoted by ΔV. The number of rows of matrix [W] equals the number of source-detector pairs, while the number of columns is the number of voxels. For a point-source, it should be noted that in equation 2.18, the expressions $G(\vec{r}_{d_i}, \vec{r}_j)$ and $U_0(\vec{r}_j, \vec{r}_{s_i})$ are represented by the same Green's function, with the only difference that they depend on different position coordinates. This solution is illustrated in figure 2.4. First, we take the fluence rate of the source at voxel position $\vec{r}_j$, $U_0(\vec{r}_j, \vec{r}_{s_i})$. In that voxel position, we assume the source $\frac{-\delta\mu_a(\vec{r})}{D} U_0(\vec{r}_j, \vec{r}_{s_i})$ whose fluence rate is described by $G(\vec{r}_{d_i}, \vec{r}_j)$. Integration over space leads us to the final expression in equation 2.17.

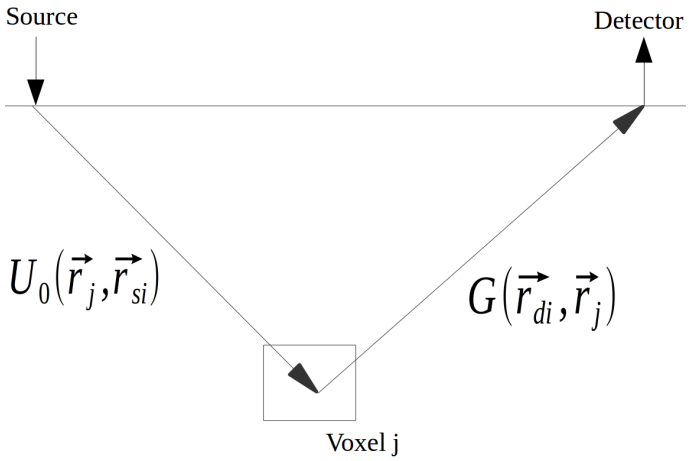

Figure 2.4: Illustration of analytical expression of Jacobian.

Eventually, however, one is interested in determining the image of absorption contrast of the tissue, represented by the perturbation vector $[\delta\mu_a(\vec{r})]$, from the fluence rate measurements $[U_{sc}(\vec{r})]$. To do that, we need to invert [W]:

$$[\delta\mu_a(\vec{r})] = [W]^{-1}[U_{sc}(\vec{r})] \tag{2.19}$$

Inverting [W] is non-trivial and computationally expensive. Among other techniques, singular-value decomposition is often used for this purpose. In this work,

the focus is on numerical determination of the Jacobian [W], not on its inversion. Constructing the proper Jacobian is one of the main steps in the tomography problem.

Chapter 3

The Monte Carlo Method

In diffuse optics, one usually employs the diffusion equation 2.2 as presented in section 2.1 to solve any problem analytically. But it was derived from the radiation transport equation 2.1 based on several assumptions, mainly that $\mu_a \ll \mu_s'$ and that the fluence rate is observed far away from sources and boundaries. We can, however, also solve the radiation transport equation numerically using the Monte Carlo method. Monte Carlo simulations are accurate and practically only limited by computational speed. In the following, I will give a thorough presentation of the Monte Carlo method as nowadays used in biomedical optics. I will focus on the open source code MBioICFO [21] that I used and modified during this work. It is based on the widely used Monte Carlo for Multi-Layered media (MCML) approach as presented by Wang *et al* [26]. MBioICFO is implemented in the object-oriented programming language C++. Since many different Monte Carlo approaches to simulate light propagation in tissue have been published over the years, I will point out differences between them and especially different ways of optimizing computational efficiency.

3.1 Implementation

Photon transport in highly scattering media is governed by random processes. Such are the path length before the photon is scattered or absorbed, the direction

into which the photon is scattered or whether it will be reflected or refracted at an index-mismatched boundary. In the Monte Carlo method, for any of these processes, a random number following a probability distribution with the expectation value equal to the physical quantity to be determined is generated. In this way, the complete trajectory of a photon in tissue can be simulated. Simulating many photons, we can estimate physical quantities such as the fluence rate at a certain position in the tissue. These quantities are ensemble-averaged and rely on averaging multiple independent samples.

In the simulation, a photon is launched at a preset location into a preset direction with an initial weight $W = 1$. With the weight parameter, it makes more sense to speak of a photon packet instead of a photon since the weight is gradually reduced during propagation. It represents the probability of the photon not to have been absorbed. To determine the step size l before the first scattering or absorption event, the first random number ξ has to be generated. Any number from 0 to 1 has equal probability to be drawn. Then, to fulfill the desired probability distribution according to the Beer-Lambert law, the step size is determined as [26]:

$$l = \frac{-\ln(\xi)}{\mu_t} \tag{3.1}$$

This is illustrated in figure 3.1, where the total attenuation coefficient $\mu_t = \mu_a + \mu_s$ is set to $\mu_t = 1\,mm^{-1}$, a realistic value for biological tissue. The curve shows the

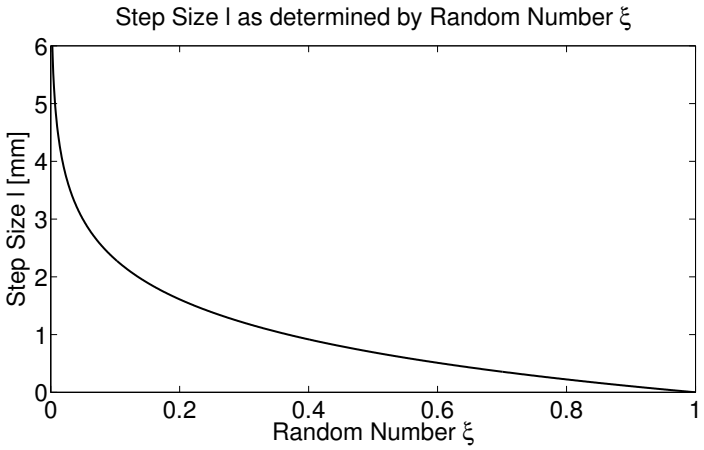

Figure 3.1: Step size l for any random number ξ generated.

step size for any random number generated. We can see that for this attenuation coefficient, most of the step sizes are less than one millimeter long.
Once the step size is set, the photon packet has to be moved. The object in which photon transport is simulated is cubic and itself divided into cubic voxels whose number and size in every direction is predefined. Each voxel consists of a predefined material with certain optical properties, namely the refractive index, the absorption coefficient, the scattering coefficient and the anisotropy factor. So the program checks whether the photon packet hits a voxel boundary when going a distance of step size l in its current direction. The visualization in figure 3.2 helps understand this better. If it hits a boundary with a different material, it might be reflected, transmitted or absorbed. Absorption happens if the neighboring material is black, so completely absorptive. Otherwise, Fresnel's formula is used to compute the internal reflectance R. It depends on the incident angle as well as on the ratio of the refractive indeces of the neighboring voxels. If a newly generated random number, again from 0 to 1, is bigger than the reflectance R, the photon packet is transmitted, otherwise it is reflected from the boundary. In both cases,

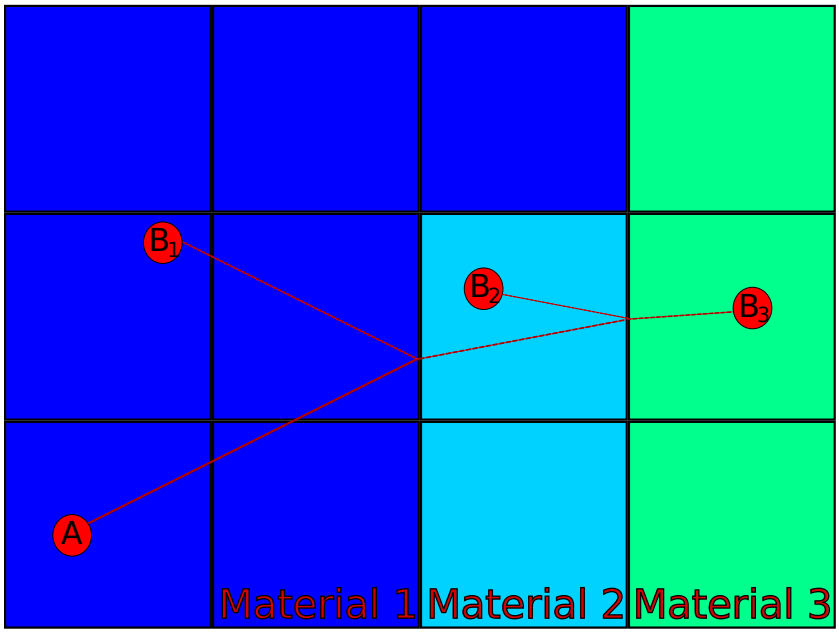

Figure 3.2: Illustration of photon step. At each interface between different materials, a random number determines whether photon is reflected or refracted.

the program assigns the photon a new direction. For transmission, refraction according to Snell's law is considered. The step size l is then reduced by the path length from the initial position to where the photon hits the boundary. After being reflected or transmitted, the photon packet might hit another boundary within the remaining step size. In that case, the same procedure is repeated until finally the step size is not big enough to reach a boundary. Then, the photon's new location lies somewhere inside a voxel. This is where the scattering happens. The photon's new direction is given by two scattering angles, the azimuthal angle, $\Psi \in [0, 2\pi)$, and the deflection angle, $\theta \in [0, \pi)$. Two independent random numbers separately determine the value of both of them. The distribution of the azimuthal angle Ψ is uniform over the entire interval. Accordingly, with the random number ξ, it is determined as:

$$\Psi = 2\pi\xi \tag{3.2}$$

The deflection angle θ, however, is described by the scattering function according to Henyey and Greenstein. The probability $p(\cos\theta)$ depends on the anisotropy g:

$$p(\cos\theta) = \frac{1 - g^2}{2(1 + g^2 - 2g\cos\theta)^{3/2}} \tag{3.3}$$

In the Monte Carlo method, we use this probability distribution to find the cosine of the deflection angle $\cos\theta$ with the random number ξ:

$$\cos\theta = \begin{cases} \frac{1}{2g}\left[1 + g^2 - \left(\frac{1-g^2}{1-g+2g\xi}\right)^2\right] & \text{if } g > 0 \\ 2\xi - 1 & \text{if } g = 0 \end{cases} \tag{3.4}$$

With these two angles, the program sets the photon's new direction and puts the step size to zero. Besides, the absorption is accounted for by dropping the weight. At each scattering event, the weight drop ΔW is calculated:

$$\Delta W = \frac{\mu_a}{\mu_t} W \tag{3.5}$$

All Monte Carlo programs usually record this deposited weight along with the coordinates of the voxel it was deposited in. In which voxels the weight of all

photon packets simulated is deposited is a measure of absorption and fluence rate at different locations. The photon's new weight W_{i+1} is then updated:

$$W_{i+1} = W_i - \Delta W \tag{3.6}$$

Note that only at scattering events, weight is dropped and absorption is accounted for. This means that when a photon packet crosses a voxel, but does not scatter in that voxel, no absorption and hence no fluence is recorded. Statistically, however, with enough photons launched, the weight drop represents well where photons have traveled and absorption has occured.

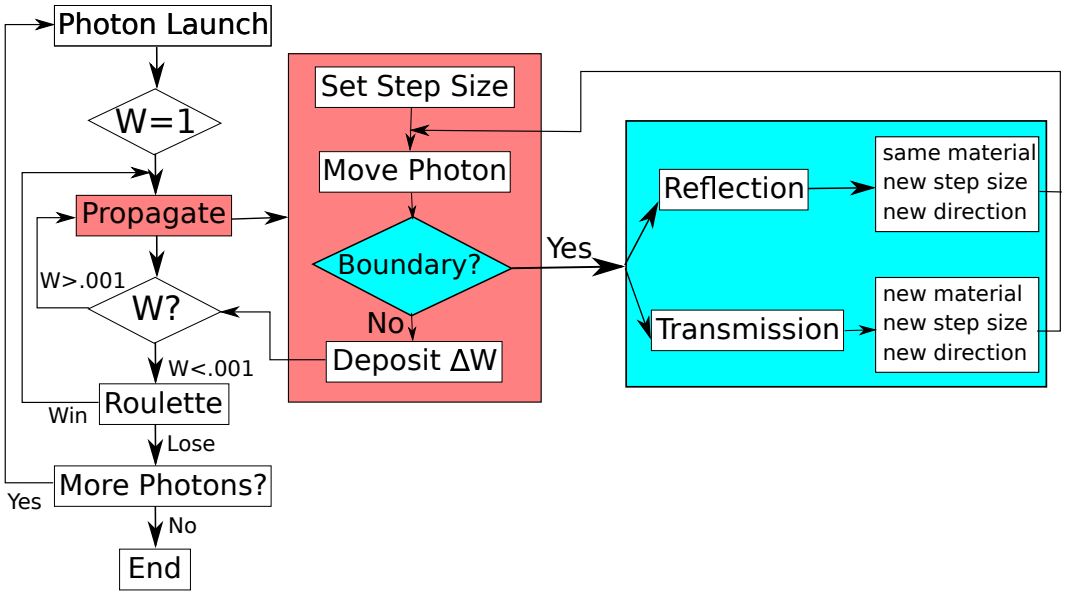

Figure 3.3: All steps in Monte Carlo method.

There are different ways in which a photon packet is terminated. It may be absorbed by a black material. These materials are typically placed at the edge of the geometry. Their absorption represents photons leaving the tissue. Additionally, a photon packet may be terminated because its weight dropped below a critical value. This value is normally in the range of 1/100 to 1/1000 of the initial weight value at photon launch [10] - in MBioICFO, it is 1/1000. Once the weight reaches this critical value, however, the photon packet is not directly terminated. For a

certain probability, usually from 1/10 to 1/100 [10], in MBioICFO 1/10, the photon survives and is assigned a new weight. The new weight is the old weight divided by the probability of survival. This ensures energy conservation. This method, called ”roulette”, makes the program more efficient since photon packets with small weights contribute only very little to the physical quantities one is interested in.
The whole procedure of simulating all the photon paths through the tissue is visualized in figure 3.3.

3.2 Input

Before execution of a simulation, one has to create an input file including several lines of information. The first line of the input file for the MBioICFO package contains the voxel size and voxel number of the geometry in each direction. Then, the user has to specify the light source, so location and direction of launched photons. This can be an isotropic point source, a collimated point source, plane wave illumination, a semicircular ring source or a ring source with variable thickness. I additionally implemented a point source with variable numerical aperture.
Furthermore, we need to specify the number and size of detectors. Any single voxel or group of voxels can serve as a detector as long as they form either a cuboid or cylindrical shape. The detector can be set to detect any photon packets passing through the detector voxels or only those that are absorbed there. For time-resolved spectroscopy (TRS), the number of time bins as well as minimum and maximum times are required.
We have to fill all voxels with materials. So we specify the number of materials we want to have in our geometry along with their optical properties. Then we fill the geometry with those materials at the desired locations. In MBioICFO, a file can optionally be prepared in Matlab to create arbitrary material shapes in the voxel grid.
Besides, we can prompt the creation of a history file for every detector and of DCS autocorrelation files. The generation of the autocorrelation functions requires specification of the laser wavelength, maximum and minimum delay times and number of time bins. Finally, on execution of the program, we specify the

number of photon packets to be simulated.
In the course of this work, I added the option to calculate the Jacobian. The time bins as well as the voxel region where the Jacobian is generated have to be specified in the input file if that option is turned on. In the next chapter, I will explain in detail the exact modifications I made to the code.

3.3 Output

The simulation of a large enough number of photon packets provides a lot of information - usually spread over several output files. All of the output files in MBioICFO, except for the history file, are saved in HDF5 [22], a data model that can store different data types in a single file. The user can import the HDF5 files into Matlab or Octave. A Matlab routine exists to conveniently import the history file as well.
One of the output files in MBioICFO is the absorption file. As mentioned above, it gives the amount of weight deposited into each voxel. So the file is a three-dimensional matrix whose size depends on the number of voxels. Knowing the absorption coefficient in the respective voxel, we can use the absorption file also to compute the fluence rate:

$$\Phi(\vec{r},t) = \frac{h\nu}{V\Delta t\mu_a(\vec{r})N_p}\sum_{i=1}^{N_p}\Delta W_i(\vec{r},t) \tag{3.7}$$

The absorption file directly provides the sum on the right-hand side, namely the sum of weights dropped by all photons in the voxel at position $\vec{r}$ at time t. Planck's constant h, the light frequency ν, the size of the time bin Δt and the number of photons N_p are considered to calculate the fluence rate.
A TRS file is generated for every detector. It contains the weight detected at each detector voxel in each time bin. Often, the history file can be used alternatively. As for the TRS files, one history file is created for each detector. For every photon detected, it contains the coordinates of the voxel it was detected in, its remaining weight and the pathlengths through every material on its way from source to detector. The history file is useful in various aspects. Its major advantage is that it

allows the calculation of physical quantities for different optical properties of the material without running the simulation more than once. As an example, to calculate the CW intensity at the detector for different absorption coefficients of the materials, we run one simulation with an absorption coefficient of zero of all the materials. With the information of the pathlengths through the materials provided by the history file, we can calculate the weights of the individual photon packets detected with the Beer-Lambert's law:

$$I = \sum_{i=1}^{N_p} W_i = \sum_{i=1}^{N_p} \exp\big(-\sum_{j=1}^{N_m} \mu_{a,j} s_{i,j}\big) \tag{3.8}$$

So in the postprocessing, we can directly substitute different absorption coefficients $\mu_{a,j}$ for any of the N_m materials without losing time waiting for more simulations to finish. The pathlengths $s_{i,j}$ are all given in the history file.
Moreover, we can use the history file to compute the electric field autocorrelation function $G_1(\tau)$ in the detector:

$$G_1(\tau) = \sum_{i=1}^{N_p} \Bigg(\exp\big(-\sum_{j=1}^{N_m} \mu_{a,j} s_{i,j}\big) \cdot \exp\big(-\sum_{j=1}^{N_m} \frac{\alpha}{3} \mu'_{s,j} s_{i,j} \langle \Delta r_j^2(\tau) \rangle \big(\frac{2\pi n_j}{\lambda_0}\big)^2\big) \Bigg) \tag{3.9}$$

This includes the factor α accounting for the ratio of scattering that happens at moving scatterers to scattering at static ones. One often assumes it to be unity. The index of refraction n_j of the respective material as well as the vacuum laser wavelength λ_0 are included. Again, one single simulation run is sufficient to compute the electric field autocorrelation function for different absorption coefficients $\mu_{a,j}$ and different mean square displacements $\langle \Delta r_j^2(\tau) \rangle$ of the scatterers. Assuming Brownian motion for example with $\langle \Delta r_j^2(\tau) \rangle = 6 D_b \tau$, different diffusion coefficients D_b can be substituted into equation 3.9 for all the N_m different materials.
Alternatively, MBioICFO provides the option to generate an autocorrelation file for the entire geometry, so that one autocorrelation function with various delay times is determined for every voxel. This, however, leads to huge output files which might be difficult to handle. Usually, especially if only the autocorrelation

in the detector is of interest, one should use the history file. Still, the autocorrelation file can be useful to determine the sensitivity $S(\tau, \vec{r}')$ at voxel position $\vec{r}'$ and source and detector at positions $\vec{r}_s$ and $\vec{r}_d$, respectively:

$$S(\tau, \vec{r}') = G_1(\tau, \vec{r}_s, \vec{r}')G_1(\tau, \vec{r}_d, \vec{r}') \tag{3.10}$$

Unfortunately, two simulations have to be run and hence two large files have to be handled. This is unpractical and ultimately we want to be able to do this in one simulation run.

3.4 Variations in the Monte Carlo Method

Most Monte Carlo simulation packages designed for biomedical optics are based on the approach outlined above. Most efforts on expanding and improving the algorithms concentrate on making the method computationally more efficient. One very effective approach is to use information from only one simulation and modify optical properties in the postprocessing. This is what I explained in the previous section with the history file. Zaccanti *et al* [19] have demonstrated a similar approach. They report a method in which the locations of all scattering events are recorded during the simulation. The temporal response when scattering or absorbing perturbations are introduced is then evaluated in the postprocessing using two scaling relationships, one both for absorption and scattering. So again, information from only one simulation run is needed.

Boas *et al* [3] have presented a simulation package "tMCimg", in which spatially varying optical properties in 3D media can be introduced to solve the forward problem. Fang *et al* [9] have extended this code to a package called "Monte Carlo eXtreme" (MCX) for parallel computing to achieve shorter runtimes. Other groups also report an acceleration of computation times by a factor of up to 10^2-10^3 by using graphics processing unit (GPU) based Monte Carlo implementations [1, 18].

An alternative to the voxelized model is a mesh-based geometry as shown by Margallo-Balbás *et al* [16] in their code "TriMC3D". To improve computational efficiency, they use a geometry based on a set of triangle meshes structured with

a space partitioning scheme.

Wang *et al* suggested using the Monte Carlo method in conjunction with diffusion theory in a hybrid model [25]. This method combines the advantage of the Monte Carlo method, accuracy, and of diffusion theory, computational efficiency, to reach up to 100 times faster computation times than the conventional Monte Carlo approach. In the hybrid model, the Monte Carlo method is only used in regions where the diffusion approximation does not hold.

Besides, there exist various techniques in the simulation of photon propagation and detection to reach better computational efficiency. One way of increasing efficiency, for example, is to split photons during their random walk [24]. This makes sense for large source-detector separations so that more split photons actually reach the detector. To conserve energy, the weight is divided on the split photons. A very similar technique is called forced detection as reported by Churmakov *et al* [5]. This method calculates the small probability that a photon packet goes directly from a scattering event to the detector and a weight is detected proportional to that probability. Various simulation packages also make use of symmetry in the simulated geometry to reduce computation time. Axial symmetry around the light source can be used for example [15].

Some groups have also addressed computing the Jacobian within the Monte Carlo method. The standard approach is to run two simulations to solve the adjoint problem, with the source at the detector position for the second simulation run. The fluence rates of both simulations in each voxel are then multiplied, giving the elements of the Jacobian. Alternatively, axial symmetry can be exploited by introducing absorption perturbations successively in voxels on a radial line and recording the thereby created change in fluence rate. Zaccanti *et al* [19] developed a code that records the location of all scattering events to plot the scattering density. While this comes close to the Jacobian, it does not properly represent it. I am aiming to implement a numerical solution that computes the correct Jacobian of any asymmetric geometry in one simulation run. One could then use this Jacobian in the inverse problem.

Chapter 4

Numerical Solution of Jacobian

For finding an implementation of the Monte Carlo method that numerically solves the Jacobian, we face two challenges. First, we have to theoretically develop an approach of how to correctly calculate the Jacobian within the existing Monte Carlo method. How can we use the information about photon transport in tissue provided by the simulation? Next, we have to ensure the developed approach is computationally efficient. This is the major limiting factor of Monte Carlo simulations. Both problems have to be solved to end up with a useful code. In the following, I devote one section to each of the two problems.

4.1 Theoretical Approach

As explained in section 2.2, each element of the Jacobian gives the differential of the fluence rate at the detector with respect to a perturbation in absorption in a specific voxel of the geometry [8]. So it is a measure of how much the fluence rate at the detector changes when the absorption coefficient in one or several voxels changes. Naturally, the Jacobian elements for voxels that many photons pass through on their way from source to detector will have a higher absolute value than those few photons pass through. The Monte Carlo method simulates the path of all photons through the tissue. This photon path information is valuable for constructing the Jacobian.
For a photon path length l_i through voxel i of absorption coefficient $\mu_{a,i}$, the prob-

ability of absorption according to Beer-Lambert's law is

$$P(absorption\ in\ voxel\ i) = 1 - \exp(-\mu_{a,i} \cdot l_i) \tag{4.1}$$

The probability of the photon to be absorbed at any point on the path from source to detector is

$$P(absorption\ anywhere) = 1 - \exp(-\sum_{i=1}^{N_V} \mu_{a,i} \cdot l_i) \tag{4.2}$$

The path length l_i will be zero for most of all N_V voxels since a photon packet usually only visits a fraction of all voxels in the geometry. The sensitivity of the fluence rate at the detector to absorption in voxel i is then for one photon:

$$J_i = \frac{1 - \exp(-\mu_{a,i} \cdot l_i)}{1 - \exp(-\sum_{i=1}^{N_V} \mu_{a,i} \cdot l_i)} \tag{4.3}$$

The task is to find a way of implementing the computation of this sensitivity into the Monte Carlo method which uses the concept of weights to account for absorption and Beer-Lambert's law.

Let us start by considering the scattering density $n_s(\vec{r_i}, t)$ as already introduced by Zaccanti *et al* [19]. It gives the density of scattering events detected photons experienced in voxel i at position $\vec{r_i}$ when detected at time t. It is normalized by the total number of photons launched at the source:

$$n_s(\vec{r_i}, t) = \frac{number\ of\ scattering\ events\ in\ voxel\ i}{total\ number\ of\ photons\ launched} \tag{4.4}$$

With the mean step size of $1/\mu_t$ between two scattering events, we can approximate the average path length l_i in each voxel as:

$$l_i = \frac{n_s(\vec{r_i}, t)}{\mu_t} \tag{4.5}$$

Again using Beer-Lambert's law, we can write the relative decay of the number of photons I_i in voxel i as:

$$I_i = I_{i-1} \exp(\frac{-\mu_a \cdot n_s(\vec{r_i}, t)}{\mu_t}) \tag{4.6}$$

This leads us to formulate the change of the number of photons when an absorption perturbation is introduced:

$$\Delta I_i = I_{i-1}(\exp(-\frac{(\mu_a + \Delta\mu_a)n_s(\vec{r_i}, t)}{\mu_t}) - \exp(\frac{-\mu_a \cdot n_s(\vec{r_i}, t)}{\mu_t})) \tag{4.7}$$

Here, we assume that $\mu_a \ll \mu_s$ and $\Delta\mu_a < \mu_a$ so that the change in the total attenuation coefficient $\mu_t = \mu_a + \mu_s$ can be neglected. For small perturbations $\Delta\mu_a$ we can make use of the first order Maclaurin series $e^x \approx 1 + x$:

$$\Delta I_i = I_{i-1}(\frac{\mu_a \cdot n_s(\vec{r_i}, t)}{\mu_t} - \frac{(\mu_a + \Delta\mu_a)n_s(\vec{r_i}, t)}{\mu_t}) = I_{i-1} \cdot (-)\frac{\Delta\mu_a}{\mu_t} n_s(\vec{r_i}, t) \tag{4.8}$$

So the sensitivity, the Jacobian element, simply reduces to:

$$J_i = -\frac{n_s(\vec{r_i}, t)}{\mu_t} \tag{4.9}$$

This, however, does not consider that the sensitivity also depends on the photon's total path length and how many other voxels a photon passes when going from source to detector as is expressed by the denominator in equation 4.3. We are looking for a way to reduce the sensitivities in voxels that are visited by photons of relatively long path lengths from source to detector. These photons will have small weights when they reach the detector. The sensitivities of those voxels mostly visited by photons of relatively short path length should be larger relative to the rest.

There are two ways to account for that. Either we multiply each value dropped by a photon packet in a voxel by the packet's remaining weight W when it is detected. Or we divide each value by the weight that was dropped on the entire path from source to detector, $1 - W$. To compare both methods, we plot the ratio of the two different factors for different remaining weights W. The ratio is:

$$Factor\ Ratio = \frac{W}{\frac{1}{1-W}} = W - W^2 \tag{4.10}$$

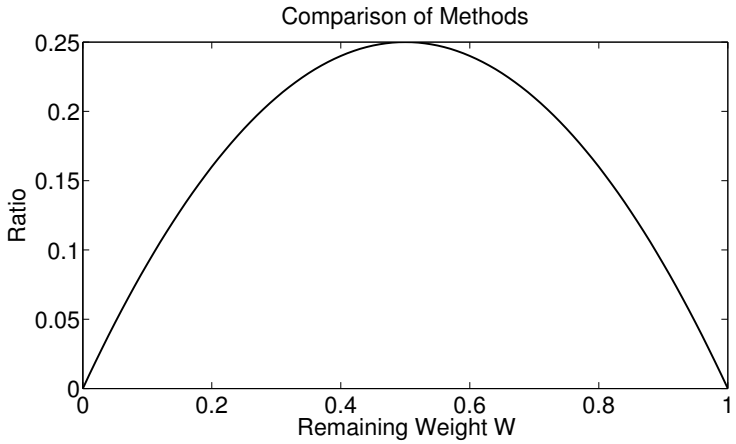

Figure 4.1: Factor of remaining weight divided by factor of division by dropped weight.

From figure 4.1 we can see that the choice of method depends on how much weight the photon packets have when they reach the detector. Both methods lead to relatively smaller sensitivity values in voxels far away from source and detector because these voxels are visited by photons that will have relatively low weight at the detector. To what extent the two methods shift the sensitivity ratio between near and far voxels, however, depends on how much weight photon packets still have when detected. If most photons have weights of less than 0.5 at the detector, the method of multiplying by the remaining weight leads to a higher difference in sensitivity values between near and far voxels than dividing by the dropped weight. The effect is reversed if photon packets arrive at the detector predominantly with weights larger than 0.5. The amount of weight photon packets have at the detector depends on the geometry, the optical properties of the materials in the geometry and the source-detector separation distance. In the configurations I worked with, most photons arrive with a weight of less than 0.5 at the detector and the method of multiplying the deposited values by the remaining weight gives better results. The Jacobian element J_i in voxel i is then computed as:

$$J_i = -\frac{1}{N_p} \sum_{j}^{N_{dp}} N_{s,j,i} \cdot \frac{1}{\mu_{t,i}} \cdot rW_j \tag{4.11}$$

All detected photons N_{dp} have a different remaining weight rW_j and a different

number of scattering events $N_{s,j,i}$ of photon j in voxel i. The total attenuation coefficient $\mu_{t,i}$ can be different in each voxel i. The division by the total photon number N_p serves as a normalization.

4.2 Implementation

In the previous section, I explained the theoretical approach to computing the Jacobian within the Monte Carlo method. Here, I want to go into more detail about how to actually implement it in the code. I acquired all the relevant programming knowledge and skills from the excellent book "A complete guide to programming in C++" by Ulla Kirch-Prinz and Peter Prinz [14].
The central quantity we are interested in is the scattering density $n_s(\vec{r_i}, t)$ of all detected photons. To extract it, we need to record all scattering events. Computationally, the challenge is that we do not know whether a photon packet will eventually be detected. So a photon's scattering events have to be recorded even though it might not be detected. If the photon packet is terminated without reaching the detector, the information about the scattering events is of no use and hence deleted. For detected photons, the location of all scattering events and time of detection has to be added to the right location of the Jacobian matrix. It is critical but non-trivial to develop a computationally efficient implementation for this process.
The code of MBioICFO is written in the object oriented programming language C++, in which each physical object like the source or the detector is defined in a class with its own specific properties. To dynamically allocate memory space to a matrix containing the location of scattering events, I implemented a class to create sparse matrices. The advantage of sparse matrices is that they only require as much memory as is really needed. So when a photon is launched, no memory is yet occupied by the sparse matrix. Only for each scattering event, the memory space is expanded as the coordinates and the total attenuation coefficient in that voxel are saved. On detection or termination of the photon, the sparse matrices occupy differently large memory space according to how many scattering events the photon experienced.
Figure 4.2 provides a visualization of this process. At each scattering event, the value $1/\mu_t$ in that voxel is written to the sparse matrix along with the voxel coor-

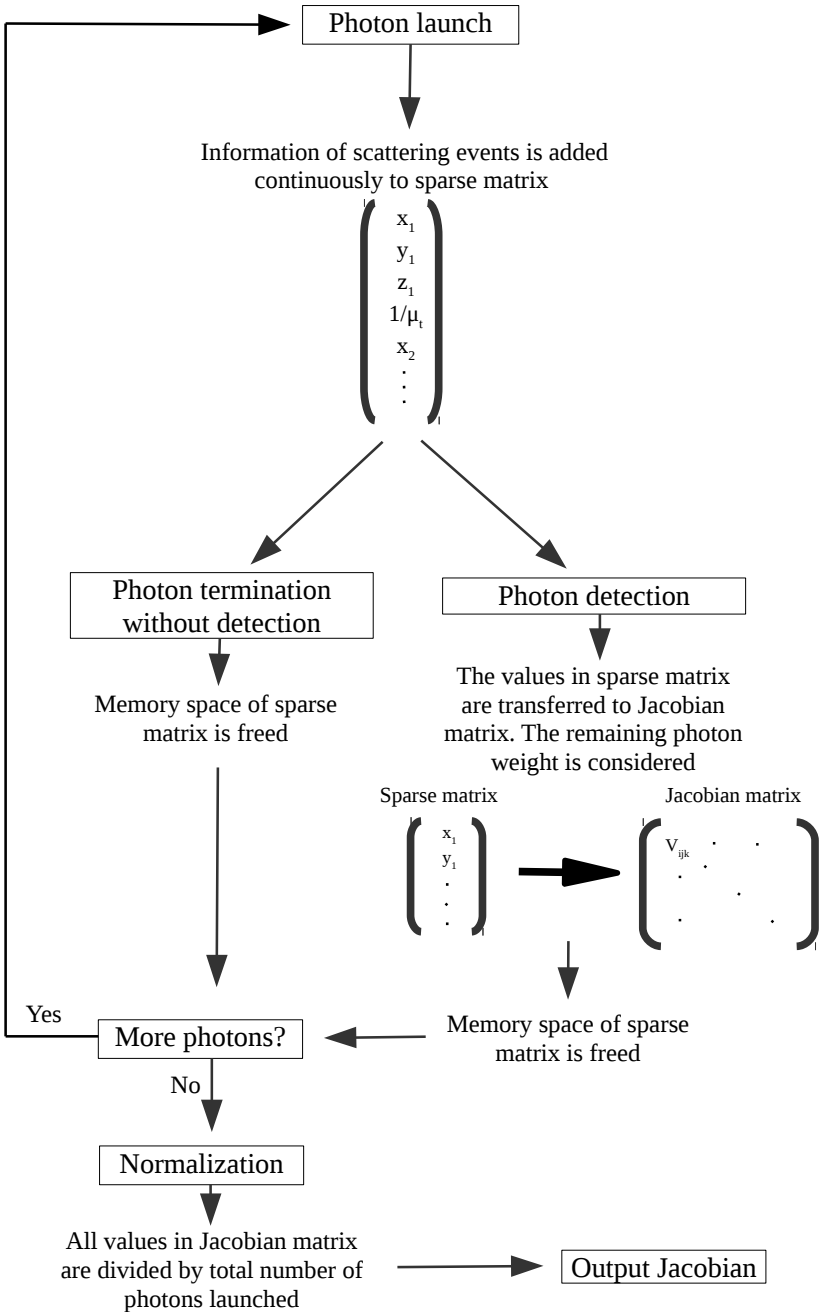

Figure 4.2: Construction of Jacobian in Monte Carlo method.

dinates. At some point, the photon is either detected or terminated without detection. In the latter case, the memory space of the sparse matrix is freed and the next photon can be recorded. At detection, the information in the sparse matrix is added to the right place in the Jacobian matrix where the remaining weight of the detected photon is considered. The Jacobian matrix has one element for each voxel and each time bin defined in the input, so its size corresponds to $N_x N_y N_z N_t$, with N_i being the number of voxels in direction i and N_t the number of time bins. Note that this size can be specified in the input, so it usually does not contain the full geometry. One can exactly specify where in the geometry and for what time bins the Jacobian is to be constructed. The voxel coordinate is read out from the sparse matrix and the time bin is assigned according to when the photon was detected. The sparse matrix's memory space is then freed and ready for the next photon. Once the program has simulated all photons, the Jacobian is normalized by dividing all its elements by the total number of photons launched. The user can then use Matlab to read out the separate file reserved for the Jacobian.

Chapter 5

Evaluation

5.1 The Jacobian in the Infinite Medium

To validate the code we compare the continuous wave analytical solution of the Jacobian in an infinite homogeneous medium according to equation 2.18 to the simulation results. The optical properties of the medium are $n = 1.33$, $\mu_a = 0.05\ mm^{-1}$, $\mu_s = 1\ mm^{-1}$ and $g = 0$. In the simulation, both the source and detector have a distance of 4 cm to all boundaries, so that we can legitimately consider the geometry as an infinite medium. The distance between source and detector is 2 cm. The result is a medium of $8cm \times 10cm \times 8cm$ divided into cubic voxels with an edge length of 0.5 mm. We are not interested in constructing the Jacobian for the entire geometry, however, but only for a smaller region around source and detector. Figure 5.1 shows the analytical solution in the infinite medium where the Jacobian values are plotted logarithmically. The values are averaged over 20 voxels, so 1 cm, in x-direction.
Figure 5.2 shows the numerical solution of the Jacobian as computed by the new implementation of the Monte Carlo method. We consider only one time bin of $10\ ns$, so it contains all relevant detected photons. The probability to detect photons at larger times is very small. Note that the plot parameters are equal to the analytical plot, the color bar for example has the same range. The 10^9 photons launched in total are a reasonable number, but we can see that there is still some statistical noise in the regions of relatively small values.

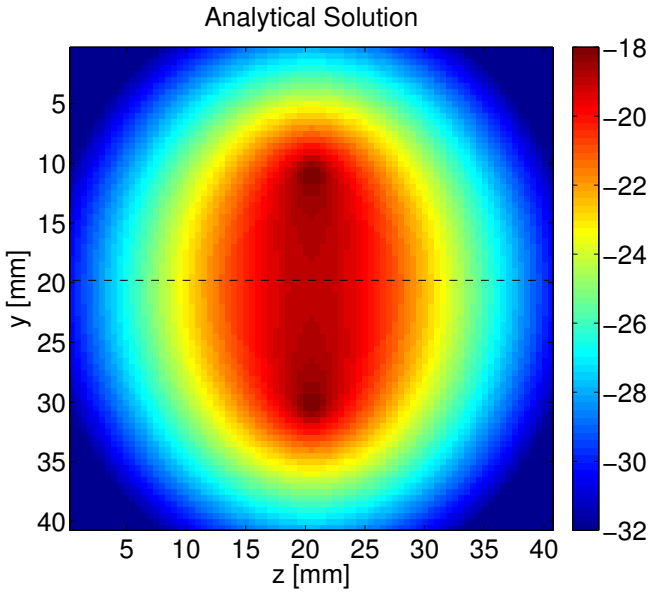

Figure 5.1: Analytical solution of Jacobian in homogeneous infinite medium. Logarithmic plot of Jacobian values averaged over 1 cm in x-direction. The dashed line shows the symmetry axis between source and detector.

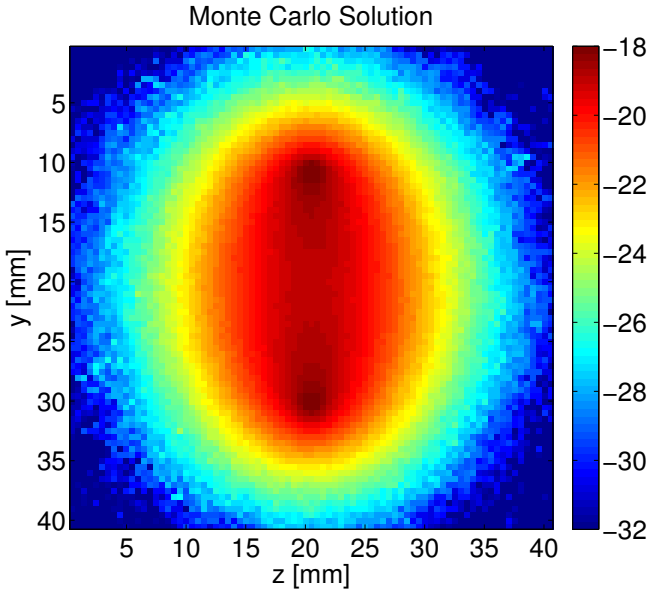

Figure 5.2: Numerical solution of Jacobian as computed by the new implementation of the Monte Carlo method. Logarithmic plot of values averaged over 1 cm in x-direction. 10^9 photons were simulated.

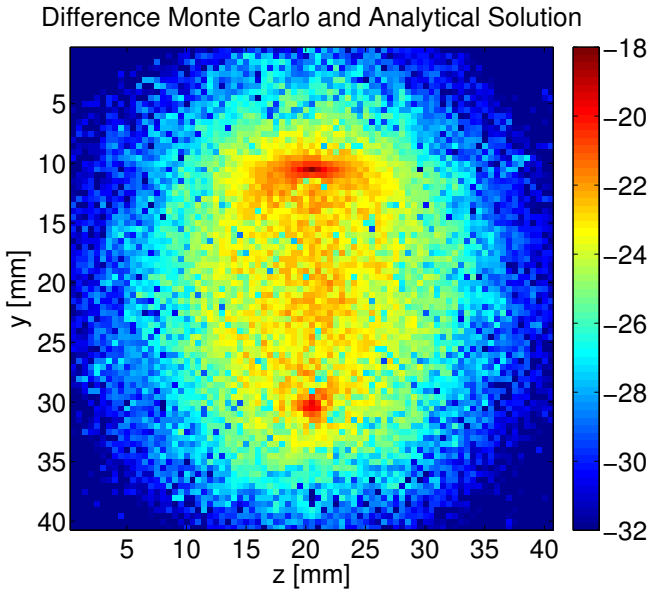

Figure 5.3: Difference between numerical and analytical solution of Jacobian plotted in logarithmic scale.

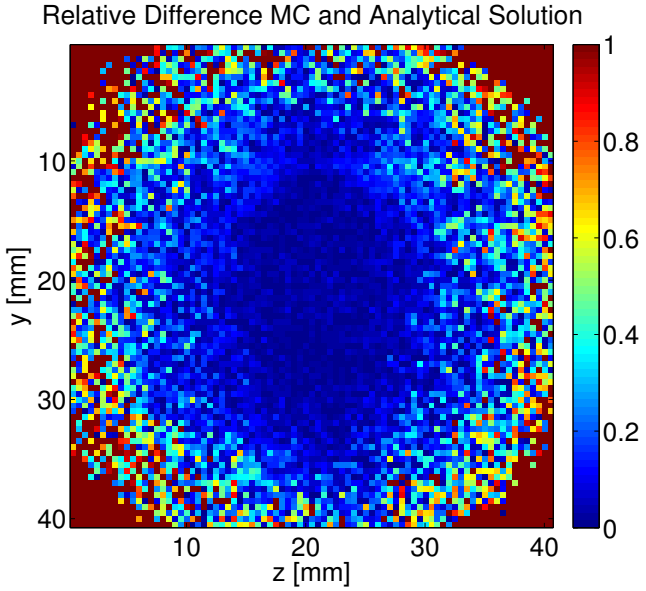

Figure 5.4: Relative difference between numerical and analytical solution of Jacobian.

To have a better comparison between analytical and numerical solution, we plot

the difference between them as shown in figure 5.3. We have to keep in mind that the analytical solution is based on the diffusion equation 2.2. That means we cannot expect reliable values close to source and detector. In the plot, these are the regions that show the highest deviation. Otherwise, there are relatively small statistic deviations, but no whole regions where one of the two solutions is consistently different. The plot of the relative difference 5.4 confirms that there is reasonable agreement in the regions much visited by the photons. The regions less visited show higher statistical noise.

To make a more precise comparison, we plot the Jacobian values along the symmetry axis between source and detector. The location of the symmetry axis is shown by the dashed line in figure 5.1. The values are again averaged over 1 cm in x-direction. In figure 5.5, the plots of the analytical and Monte Carlo solutions show good agreement. Figure 5.6 shows the normalized natural logarithm of the Jacobian values which is more suitable to compare the exponential decays. Apart from the statistical noise fluctuations in the Monte Carlo solution, the two graphs decay identically. Note that these Jacobians consist of negative values. For illustrative reasons, we consider their absolute values in the plots.

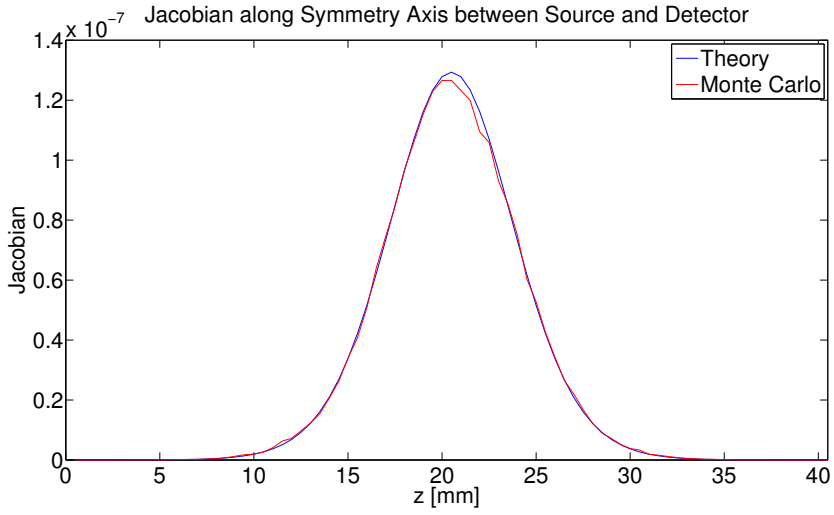

Figure 5.5: Comparison between the numerical and the analytic solution.

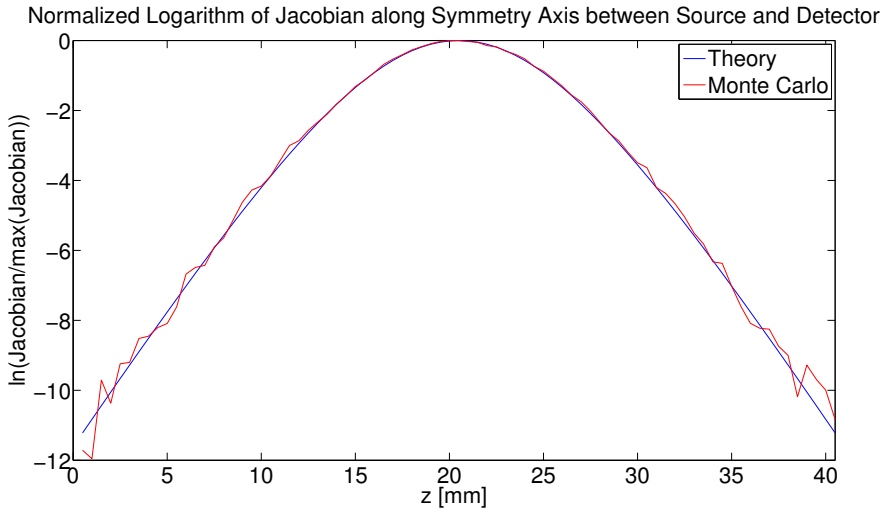

Figure 5.6: Normalized logarithm of numerical and analytical solution.

5.2 The Jacobian in the Semi-infinite Medium

As a further validation of the method, we test it in the semi-infinite medium. The introduction of the boundary might reveal possible shortcomings in the approach. For a symmetric configuration comparable to the analytical solution, we place the source and detector a distance $l_{tr} = 1/\mu_s'$ from the boundary. Source-detector separation is 2 cm again and the voxels are cubic with edge length 0.5 mm. The optical properties of the medium are $n = 1.33$, $\mu_a = 0.05\ mm^{-1}$, $\mu_s = 1\ mm^{-1}$ and $g = 0$. The material on the other side of the boundary is air with an index of refraction $n = 1.00$, so the boundary is index-mismatched. To set up the analytical solution, we take into account the corresponding boundary conditions 2.9 with the proper effective reflection coefficient R_{eff}. In the following figures, photons detected at all times are taken into account. For a better visualization, we take the natural logarithm of the Jacobian in the plots. All values are again averaged over 1 cm in x-direction. We simulated 10^9 photons.

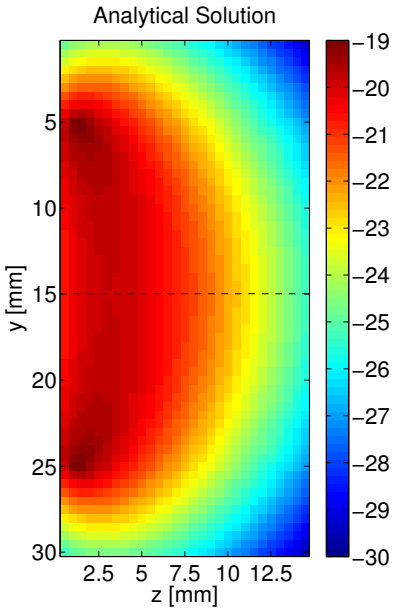

Figure 5.7: Analytical solution plotted in logarithmic scale.

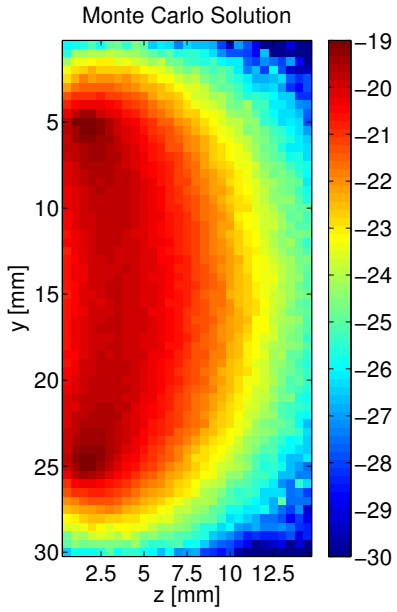

Figure 5.8: Monte Carlo solution plotted in logarithmic scale.

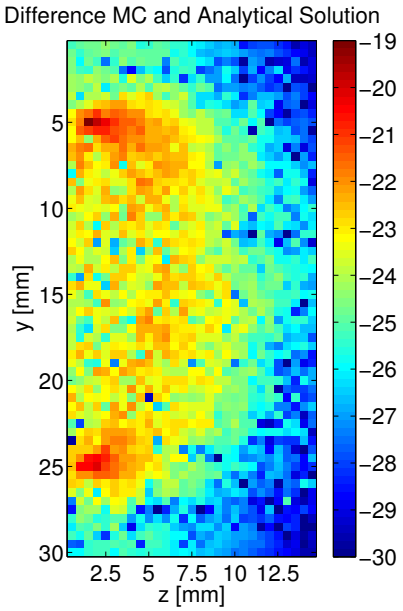

Figure 5.9: Absolute difference plotted in logarithmic scale.

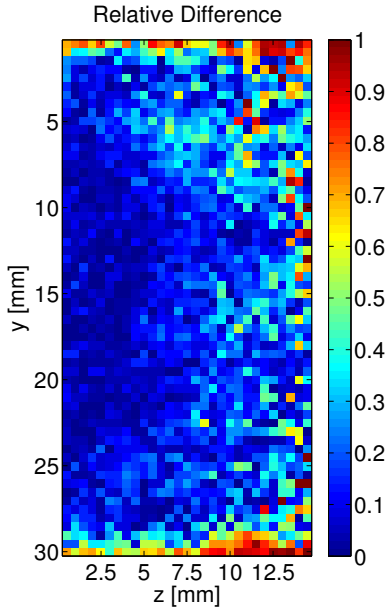

Figure 5.10: Relative difference.

The plot of the relative difference between the Monte Carlo and the analytical solution 5.10 suggests good agreement of the two. As for the infinite medium, the relative difference is only large for regions sparsely visited by photons. Note that we are directly comparing Monte Carlo and analytical solution without taking any normalization measures. For a more detailed comparison, we plot the Jacobian along the symmetry axis between source and detector as indicated by the dashed line in figure 5.7. Figure 5.11 shows the unnormalized Jacobian along the symmetry axis averaged over 1 cm in x-direction. Additionally, figure 5.12 shows the natural logarithm of the Jacobian along the same line, also not normalized. So even without normalization measures, absolute values and decay of the Jacobian show reasonable agreement. We can reduce the statistical noise fluctuations in the Monte Carlo solution by simulating an even higher number of photons. For this present study, however, limited computational power did not allow that.

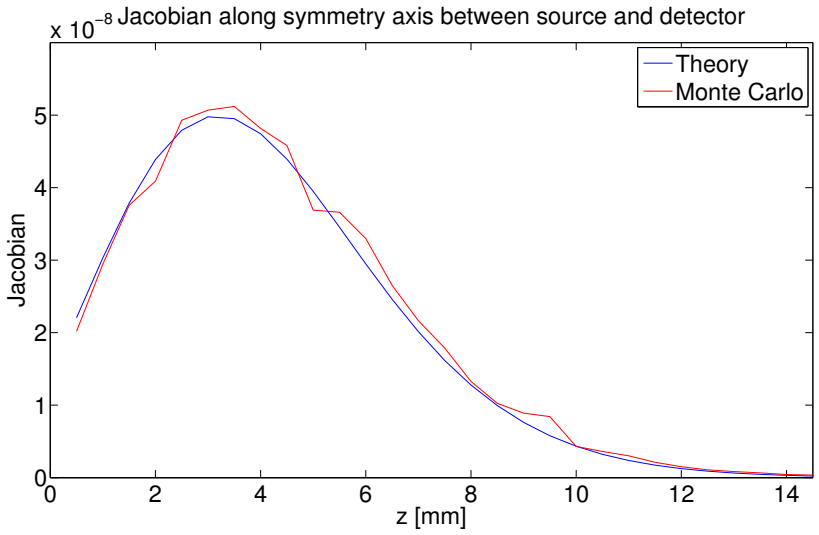

Figure 5.11: Comparison between the numerical and the analytic solution.

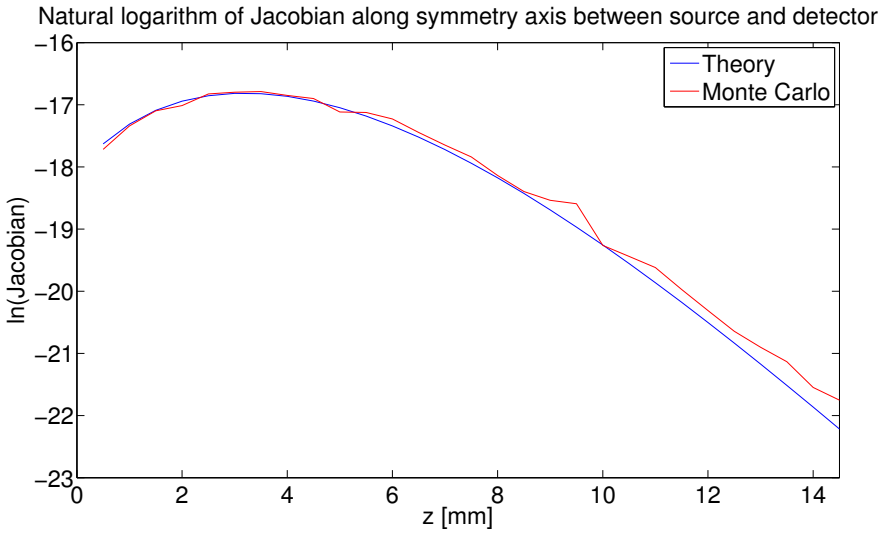

Figure 5.12: Normalized logarithm of numerical and analytic solution.

5.3 The Jacobian at Different Detection Times

As mentioned earlier, the implementation developed in the course of this work also allows to find the Jacobian for different time bins of detection. Depending on when a photon is detected, the sensitivities in different regions of the material vary. So looking at the detector signal at different times, we obtain different depth sensitivities.

The following figures show the Jacobian in five time bins of the simulation in the semi-infinite medium as presented above. In general, we can freely specify the maximum time and number of time bins in the input. The time bins shown in the figures represent the time when most photons are detected. Clearly, the figures show that for later detection times, the sensitivity is much higher at larger depths and vice versa for early detection times. All five figures are plotted with the same colorscale as depicted by the colorbar in figure 5.13. The plots again show the natural logarithm of the Jacobian averaged over 1 cm in x-direction.

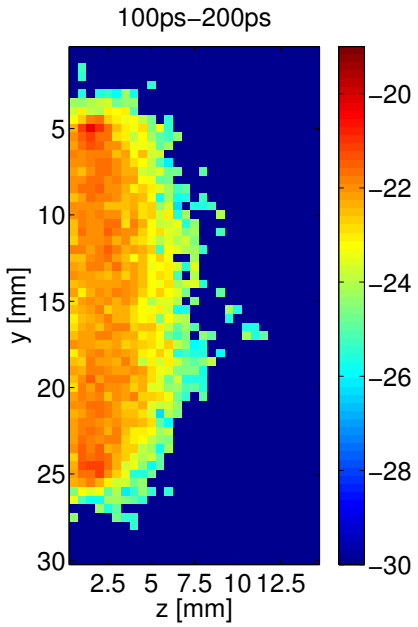

Figure 5.13: First time bin plotted in logarithmic scale.

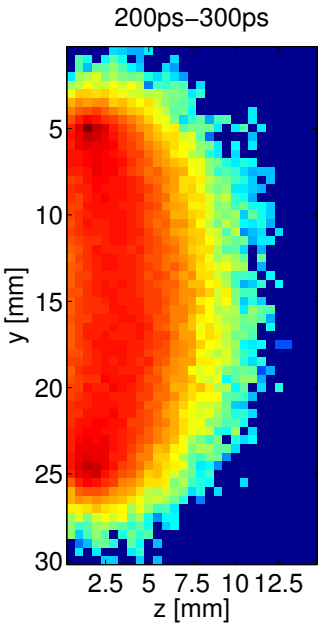

Figure 5.14: Second time bin plotted in logarithmic scale.

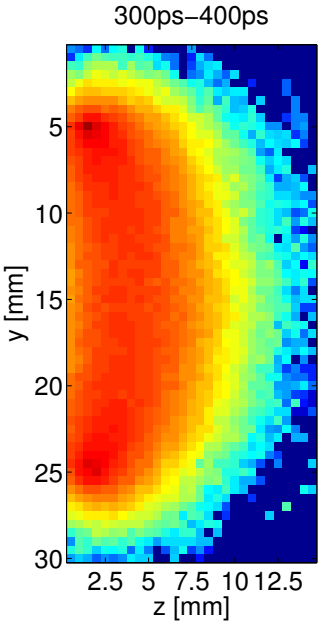

Figure 5.15: Third time bin plotted in logarithmic scale.

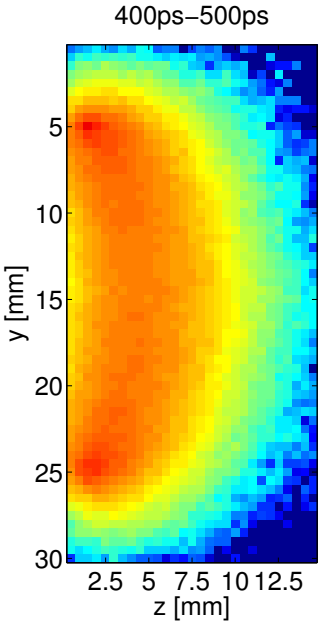

Figure 5.16: Fourth time bin plotted in logarithmic scale.

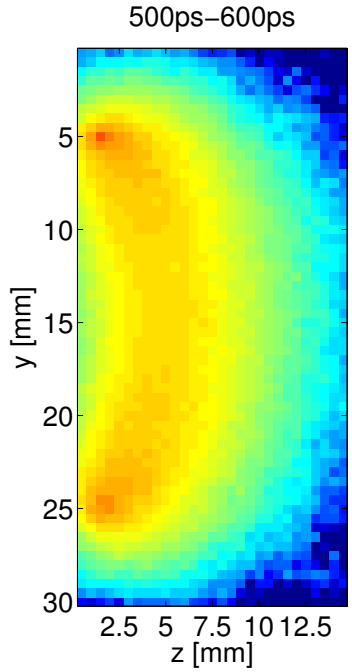

Figure 5.17: Fifth time bin plotted in logarithmic scale.

5.4 The Jacobian in MRI of Human Head

Figure 5.18 shows an anatomical magnetic resonance image (MRI) of a human head [1]. The different tissue types are the skull, the cerebrospinal fluid (CSF) and the gray/white matter. To demonstrate the utility of the method developed, we compute the sensitivities at different detection times in this MRI geometry. This shows a potential application of the work presented.

In diffuse optical tomography (DOT), several source-detector pairs are placed on the head to measure physiologically relevant variations of the optical properties in the brain [7]. Traditionally, the challenge has been to get enough depth sensitivity to reach the gray and white matter tissue of interest in the brain. For the simulation illustrated in the figures below, we choose a relatively large source-detector separation distance of 3.4 cm to achieve a larger depth sensitivity. The collimated source on the head surface that would be used in the measurement is approximated

[1] Data courtesy of Yodh lab at University of Pennsylvania

Materials:

1: μ_s = 0.81 mm^{-1} , μ_a = 0.0115 mm^{-1}

2: μ_s = 0.01 mm^{-1} , μ_a = 0.0017 mm^{-1}

3: μ_s = 1.16 mm^{-1} , μ_a = 0.017 mm^{-1}

4: μ_s = 1.16 mm^{-1} , μ_a = 0.017 mm^{-1}

g = 0.01, n = 1.33 for all of them

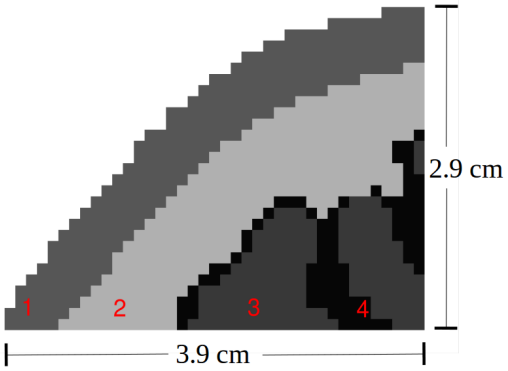

Figure 5.18: Image from MRI measurements of a human head. Material 1 is the skull, material 2 the cerebrospinal fluid (CSF) and material 3 and 4 are grey/white matter.

by an isotropic source at a depth of approximately $1/\mu_s'$. The detector is also located at that depth for symmetry reasons. The voxels in the geometry are cubic with an edge length of 1 mm. 10^9 photons were simulated. Figures 5.19 to 5.26 show the Jacobian in the MRI geometry for different detection times. The colorbar in figure 5.19 is valid for the plots of all time bins and gives the natural logarithm of the Jacobian.

As pointed out in chapter 3, the Monte Carlo method is important to determine photon transport in media in which the diffusion approximation (see section 2.1) does not hold. It can be seen in figure 5.18 that the diffusion approximation is clearly violated in the CSF, since the absorption coefficient μ_a is almost a fifth of the scattering coefficient μ_s and the reduced scattering coefficient μ_s' is even a bit smaller than μ_s. So this is a good example where the diffusion equation 2.2 cannot be used and, accordingly, we have to rely on Monte Carlo solutions.

In the figures below, the Jacobian looks very noisy in the CSF. This has mainly two reasons. First of all, as opposed to the solutions presented in the infinite and semi-infinite medium, the Jacobian is shown for only one layer of voxels because the location of tissue types in other layers is obviously different. So we can only take into account photons scattered in this layer. Additionally, the scattering coefficient is very small, two orders of magnitude smaller than in the brain. So there are fewer scattering events contributing to the construction of the Jacobian (see section 4). This leads to more noise. Noise is reduced by a shorter source-

detector separation at the cost of smaller depth sensitivity.
We can see that there is higher depth sensitivity for larger detection times. While the signal at the detector in the time span from 200 ps to 300 ps is mainly unaffected by the optical properties in the brain, the signal measured for example at 600 ps to 700 ps does contain information of the brain region. We also see how the sensitivities are lower in the 900 ps to 1000 ps time bin, because few photons are detected this late.

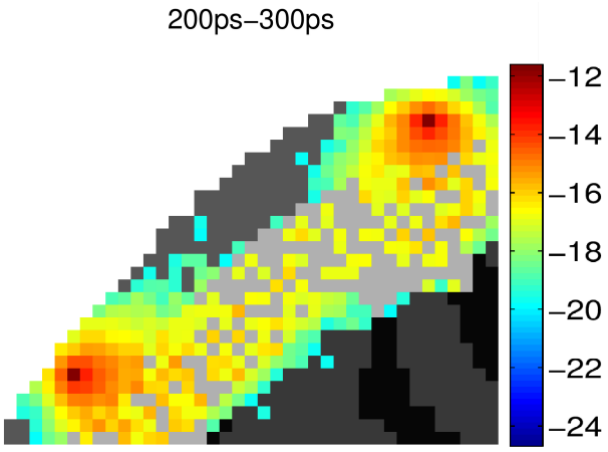

Figure 5.19: Jacobian in first time bin plotted in logarithmic scale. Source-detector separation is 3.4 cm. 10^9 photons were simulated.

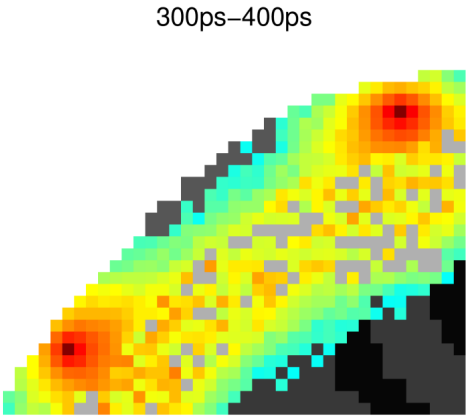

Figure 5.20: Second time bin plotted in logarithmic scale.

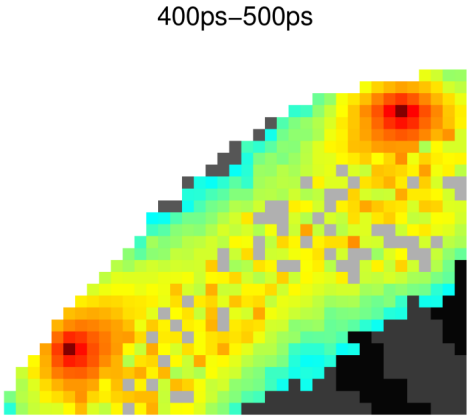

Figure 5.21: Third time bin plotted in logarithmic scale.

These figures illustrate the value of the code to visualize sensitivities in heterogeneous media at different detection times. The findings from this simulation also agree with a similar study conducted by Boas *et al* [3].

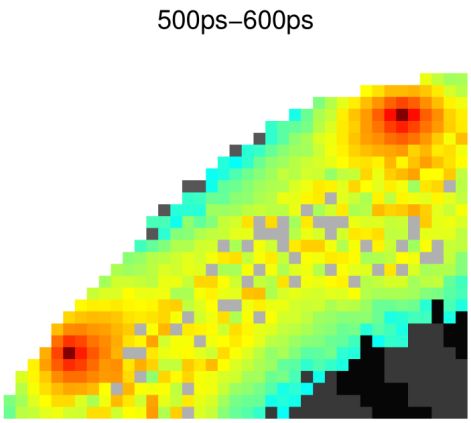

Figure 5.22: Fourth time bin plotted in logarithmic scale.

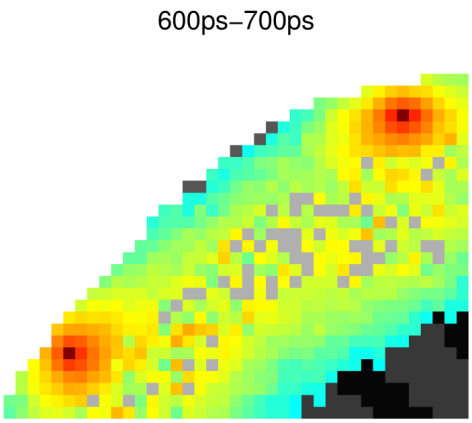

Figure 5.23: Fifth time bin plotted in logarithmic scale.

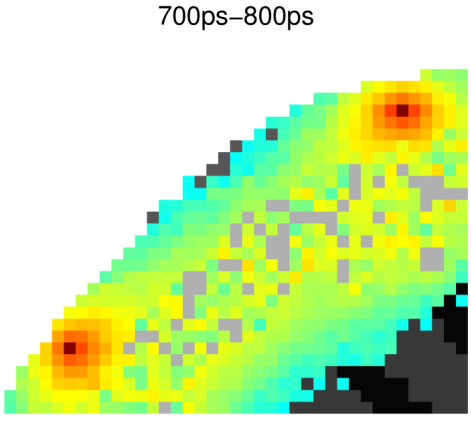

Figure 5.24: Sixth time bin plotted in logarithmic scale.

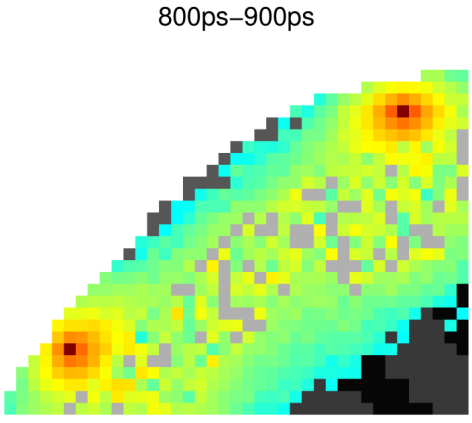

Figure 5.25: Seventh time bin plotted in logarithmic scale.

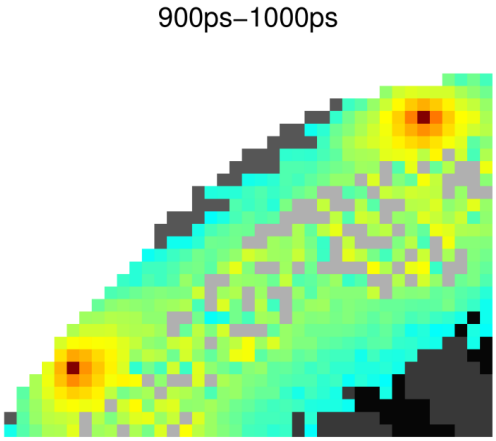

Figure 5.26: Eighth time bin plotted in logarithmic scale.

Chapter 6

Conclusion

In this work, I have presented an extension of the Monte Carlo method for diffuse optics. The existing Monte Carlo method and its importance was thoroughly introduced. I explained the significance of finding the sensitivity matrix, also called Jacobian, for any arbitrary geometry to solve the inverse problem and determine a structural image of the tissue. An approach to numerically determine the Jacobian for time-resolved spectroscopy (TRS) in one simulation run was laid out. The method was verified in the infinite as well as in the semi-infinite medium. Besides, I have shown the capability of the program to generate the sensitivities corresponding to different detection times. The implementation has proven to be computationally efficient and the results show good agreement with the analytical solutions for the number of photon packets simulated.

I have further demonstrated the power of using the new implementation of MBioICFO developed in this work to determine the sensitivity matrix in any heterogeneous medium at different detection times. Data from MRI measurements can be read in and used as the geometry.

The advancement of the Monte Carlo method can stimulate and enhance the power of diffuse optics technology in general. The numerical simulations provide a means to learn more about photon transport in tissue and as the implementations become more computationally efficient, the method becomes even more useful. Hopefully, the Monte Carlo method contributes to making diffuse optics technology a valuable tool in the hospital.

Bibliography

[1] Erik Alerstam, Tomas Svensson, and Stefan Andersson-Engels. Parallel computing with graphics processing units for high-speed monte carlo simulation of photon migration. *Journal of biomedical optics*, 13(6):060504–060504, 2008.

[2] David Boas, Dana H Brooks, Eric L Miller, Charles DiMarzio, Misha Kilmer, Richard J Gaudette, Quan Zhang, et al. Imaging the body with diffuse optical tomography. *Signal Processing Magazine, IEEE*, 18(6):57–75, 2001.

[3] David Boas, J Culver, J Stott, and A Dunn. Three dimensional monte carlo code for photon migration through complex heterogeneous media including the adult human head. *Optics express*, 10(3):159–170, 2002.

[4] Cecil Cheung, Joseph P Culver, Kasushi Takahashi, Joel H Greenberg, and AG Yodh. In vivo cerebrovascular measurement combining diffuse near-infrared absorption and correlation spectroscopies. *Physics in medicine and biology*, 46(8):2053, 2001.

[5] D Yu Churmakov, Vladimir Leonidovich Kuz'min, and Igor'Vladislavovich Meglinski. Application of the vector monte-carlo method in polarisation optical coherence tomography. *Quantum Electronics*, 36(11):1009–1015, 2006.

[6] Joseph P Culver, Turgut Durduran, Daisuke Furuya, Cecil Cheung, Joel H Greenberg, and AG Yodh. Diffuse optical tomography of cerebral blood

flow, oxygenation, and metabolism in rat during focal ischemia. *Journal of cerebral blood flow & metabolism*, 23(8):911–924, 2003.

[7] Joseph P Culver, Andrew M Siegel, Jonathan J Stott, and David A Boas. Volumetric diffuse optical tomography of brain activity. *Optics letters*, 28(21):2061–2063, 2003.

[8] Turgut Durduran, Regine Choe, WB Baker, and AG Yodh. Diffuse optics for tissue monitoring and tomography. *Reports on Progress in Physics*, 73(7):076701, 2010.

[9] Qianqian Fang and David A Boas. Monte carlo simulation of photon migration in 3d turbid media accelerated by graphics processing units. *Optics express*, 17(22):20178–20190, 2009.

[10] Ingemar Fredriksson. Quantitative laser doppler flowmetry. 2009.

[11] George M Hale and Marvin R Querry. Optical constants of water in the 200-nm to 200-μm wavelength region. *Applied optics*, 12(3):555–563, 1973.

[12] Avinash C.. Kak and Malcolm Slaney. *Principles of computerized tomographic imaging*. Society for Industrial and Applied Mathematics, 2001.

[13] Joseph F Kelleher. Pulse oximetry. *Journal of clinical monitoring*, 5(1):37–62, 1989.

[14] Ulla Kirch-Prinz and Peter Prinz. *A Complete Guide to Programming in C++*. Jones & Bartlett Learning, 2002.

[15] Marcus Larsson, Wiendelt Steenbergen, Tomas Stro, et al. Influence of optical properties and fiber separation on laser doppler flowmetry. *Journal of Biomedical Optics*, 7(2):236–243, 2002.

[16] Eduardo Margallo-Balbás and Patrick J French. Shape based monte carlo code for light transport in complex heterogeneous tissues. *Optics express*, 15(21):14086–14098, 2007.

[17] Scott Prahl et al. Optical absorption of hemoglobin. *Oregon Medical Laser Center, http://omlc. ogi. edu/spectra/hemoglobin/index. html*, 15, 1999.

[18] Nunu Ren, Jimin Liang, Xiaochao Qu, Jianfeng Li, Bingjia Lu, and Jie Tian. Gpu-based monte carlo simulation for light propagation in complex heterogeneous tissues. *Optics express*, 18(7):6811–6823, 2010.

[19] Angelo Sassaroli, Costantino Blumetti, Fabrizio Martelli, Lucia Alianelli, Daniele Contini, Andrea Ismaelli, and Giovanni Zaccanti. Monte carlo procedure for investigating light propagation and imaging of highly scattering media. *Applied optics*, 37(31):7392–7400, 1998.

[20] John C Schotland, John C Haselgrove, and John S Leigh. Photon hitting density. *Applied optics*, 32(4):448–453, 1993.

[21] M.B. Taisuke, A. Pau Giannoula, and Turgut Durduran. Mbio: a comprehensive monte-carlo package for diffuse correlation spectroscopy/tomography, 2013. European Conferences on Biomedical Optics.

[22] The HDF Group. Hierarchical Data Format, version 5, 1997-NNNN. http://www.hdfgroup.org/HDF5/.

[23] RLP Van Veen, Henricus JCM Sterenborg, Antonio Pifferi, Alessandro Torricelli, Ekaterine Chikoidze, and Rinaldo Cubeddu. Determination of visible near-ir absorption coefficients of mammalian fat using time-and spatially resolved diffuse reflectance and transmission spectroscopy. *Journal of biomedical optics*, 10(5):054004–054004, 2005.

[24] Eric Veach. *Robust monte carlo methods for light transport simulation*. PhD thesis, Stanford University, 1997.

[25] Lihong Wang and Steven L Jacques. Hybrid model of monte carlo simulation and diffusion theory for light reflectance by turbid media. *JOSA A*, 10(8):1746–1752, 1993.

[26] Lihong Wang, Steven L Jacques, and Liqiong Zheng. Mcml-monte carlo modeling of light transport in multi-layered tissues. *Computer methods and programs in biomedicine*, 47(2):131–146, 1995.

[27] Lihong V Wang and Hsin-i Wu. *Biomedical optics: principles and imaging*, chapter 1: Motivation for optical imaging. John Wiley & Sons, 2012.

[28] Lihong V Wang and Hsin-i Wu. *Biomedical optics: principles and imaging*, chapter 5: Radiative Transfer Equation and Diffusion Theory. John Wiley & Sons, 2012.